**1.安装gcc4.7**

**参考链接：**

**Caffe+MATLAB2014a+ubuntu14.04**

[**http://www.th7.cn/system/lin/201501/80713.shtml**](http://www.th7.cn/system/lin/201501/80713.shtml)

Ubuntu14.04自带的gcc版本是4.8，MATLAB2014b支持的最高版本为4.7x。因此，需要安装gcc4.7，并给gcc降级

在终端执行gcc 4.7的安装命令：

sudo apt-get install gcc-4.7 g++-4.7 g++-4.7-multilib gcc-4.7-multilib

在终端执行以下系统gcc降级命令：

sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-4.7 100

（此电脑好像没有g++4.8，所以下面这句没执行）

sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-4.8 50

sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-4.7 100

sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-4.8 50

sudo update-alternatives --install /usr/bin/cpp cpp-bin /usr/bin/cpp-4.7 100

sudo update-alternatives --install /usr/bin/cpp cpp-bin /usr/bin/cpp-4.8 50

验证gcc-4.7是否安装并成为系统的默认版本：

gcc -v

**2.安装相关依赖项**

sudo apt-get install libprotobuf-dev libleveldb-dev libsnappy-dev libopencv-dev libhdf5-serial-dev protobuf-compiler

sudo apt-get install libboost-all-dev

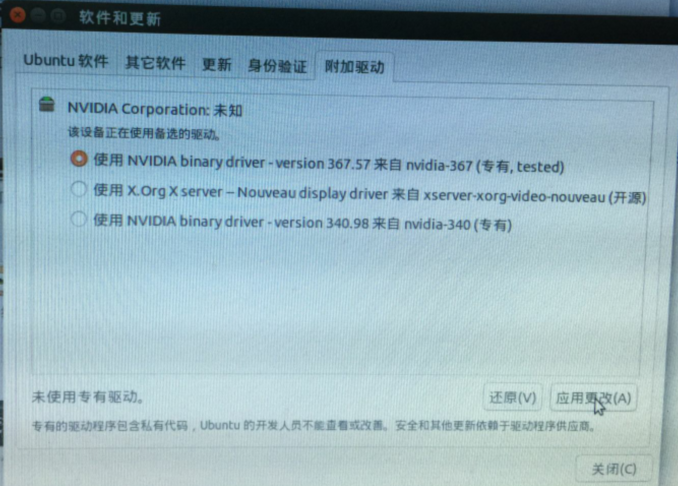
sudo apt-get install libgflags-dev libgoogle-glog-dev liblmdb-dev

sudo apt-get install libopenblas-dev liblapack-dev libatlas-base-dev

（依赖项安装完后不知道为啥g++-4.8也被安装了，所以上面那句没执行的也执行了）

**3.安装NVIDIA驱动**

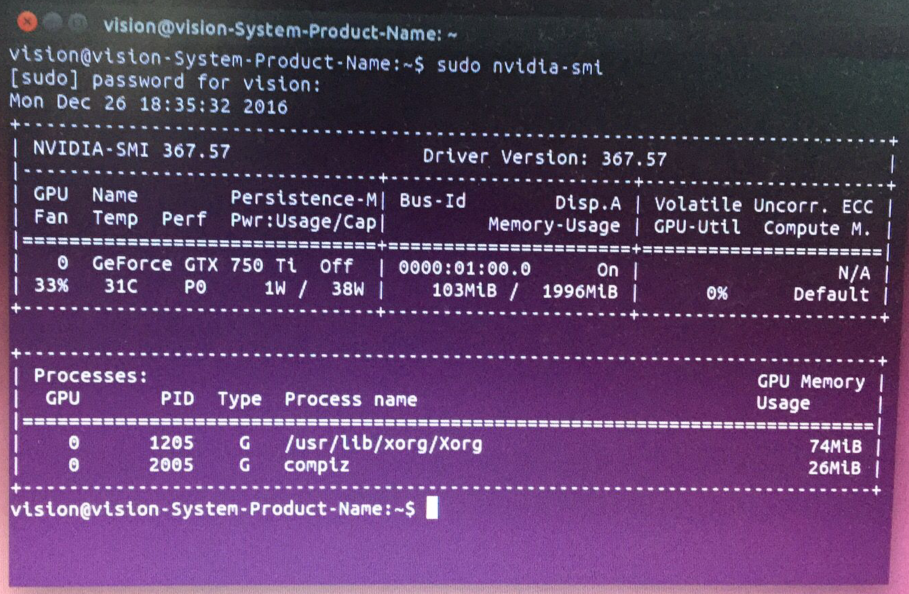
直接在ubuntu系统设置，软件和更新里面，选择中国的服务器源刷新之后，点击附加驱动选项，在Nvidia Corporation选择367(强迫症必须安装最新的)，然后点击应用更改，下载安装完之后重启



安装完成之后输入以下指令进行验证：

sudo nvidia-smi

若列出了GPU的信息列表则表示驱动安装成功。

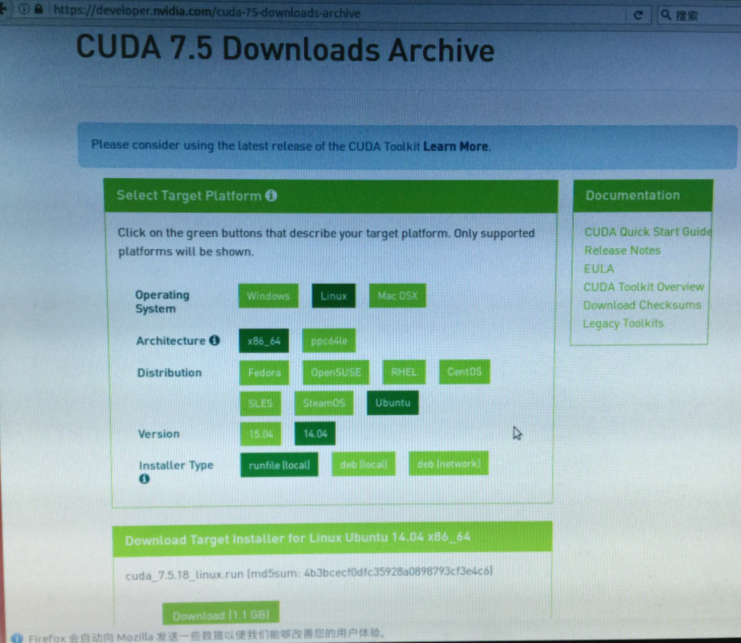


**4.** **安装CUDA**

CUDA是NVIDIA的编程语言平台，想使用GPU就必须要使用cuda。

（1）下载CUDA

首先在官网上(https://developer.nvidia.com/cuda-downloads)下载CUDA：

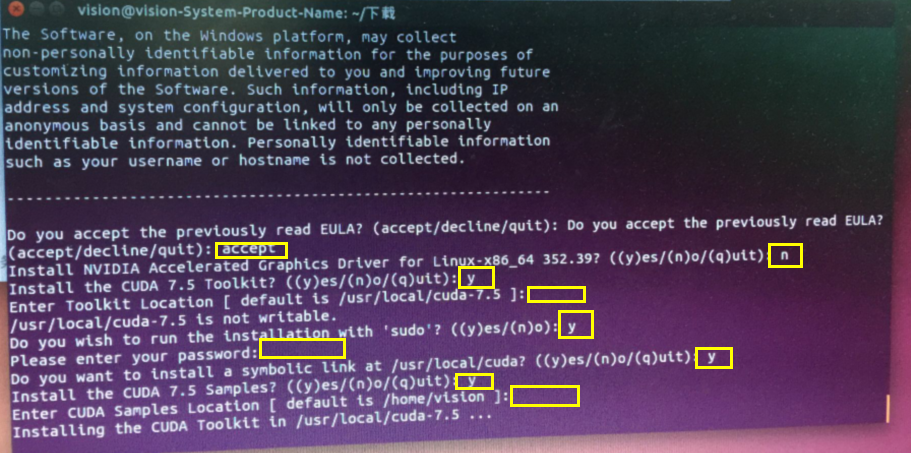


（2）安装cuda7.5

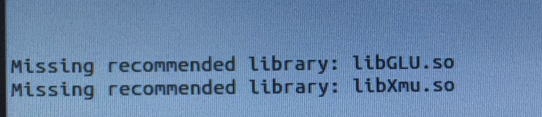
以文件名为cuda.run为例，终端下输入

sh cuda.run –override

启动安装程序，此处有大量的条款，一路空格到最后 输入accept，依次输入y回车，然后n(不安装显卡驱动)，然后一路y回车，有一个地方需要输入密码，还有两个地方确认安装路径，直接回车即可，完成安装，默认安装路径是/usr/local



安装完成，提示缺少一些库



就安装缺少的库

apt-get install libglu1-mesa libxmu-dev libglu1-mesa-dev freeglut3-dev

注意包名不要写错了，如libglu后面是数字1，不是字母l

这个因为网络问题，可能要安装很久。

还是有线好点

（3）设置环境变量

打开~/.bashrc文件：

sudo vim ~/.bashrc

将以下内容写入到~/.bashrc尾部：

export PATH=/usr/local/cuda-7.5/bin${PATH:+:${PATH}}

export LD\_LIBRARY\_PATH=/usr/local/cuda-7.5/lib64${LD\_LIBRARY\_PATH:+:${LD\_LIBRARY\_PATH}}

保存退出（按Esc，输入命令:Wq,表示保存退出）

然后刷新

输入：



（4）测试cuda

命令行回到最初文件夹，运行

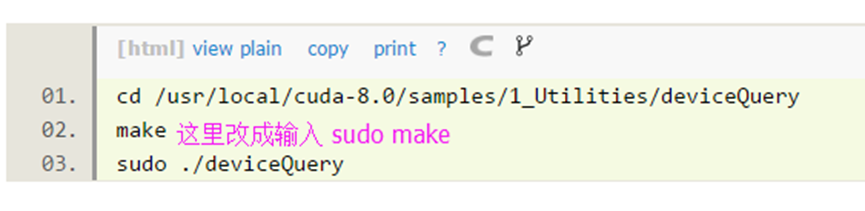
结果如下：

（5）测试cuda的sample

cd /usr/local/cuda-7.5/samples/1\_Utilities/deviceQuery

sudo make

sudo ./deviceQuery

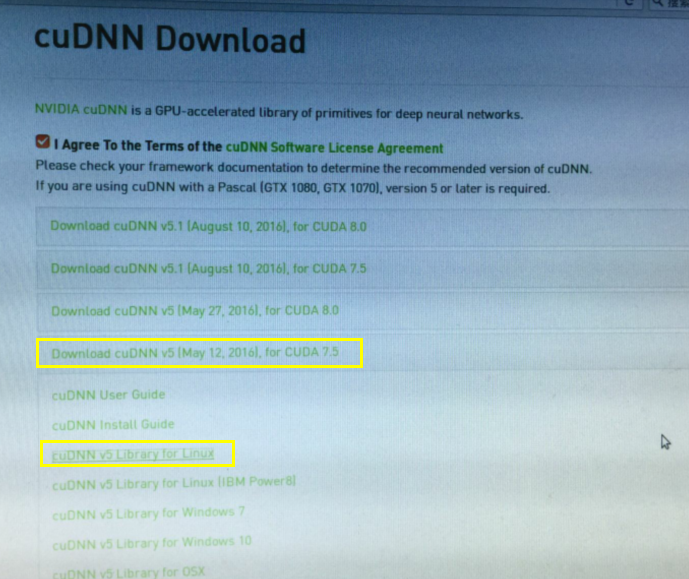


如果显示的是一些关于GPU的信息，则说明安装成功了。(最后显示是ＰＡＳＳ不是Faile!)

**5.安装cudnn5.0（对应cuda7.5版本）**

1. 下载cudnn5.0

下载链接：<https://developer.nvidia.com/rdp/cudnn-download>

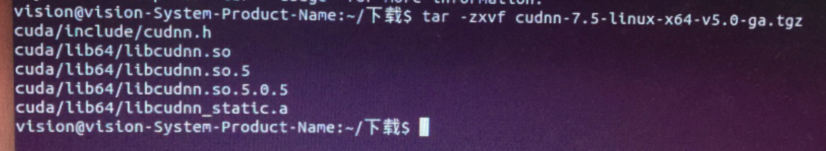


1. 安装cuDNN

安装cuDNN比较简单，解压后把相应的文件拷贝到对应的CUDA目录下即可：

解压：

tar –zxvf cudnn-7.5-linux-x64-v5.0-ga.tgz



cd进入cuDNN5.0解压之后的include目录：

cd cuda/include

在命令行进行如下操作

sudo cp cudnn.h /usr/local/cuda/include #复制头文件

再将cd进入lib64目录下的动态文件进行复制和链接：



sudo cp lib\* /usr/local/cuda/lib64/ #复制动态链接库

cd /usr/local/cuda/lib64/

（以下命令要根据不同版本的cudnn进行修改，不然后面caffe安装会有很多这边的问题）

sudo rm -rf libcudnn.so libcudnn.so.5 #删除原有动态文件

sudo chmod u=rwx,g=rx,o=rx libcudnn.so.5.0.5 #修改文件权限

sudo ln -s libcudnn.so.5.0.5 libcudnn.so.5 #生成软衔接

sudo ln -s libcudnn.so.5 libcudnn.so #生成软链接

更新链接库：

**sudo ldconfig**

测试cuda有很多方法,下面的测试方法比较全面，如果报错，最好全部google改过来

cd /home/vision/NVIDIA\_CUDA-7.5\_Samples

sudo make all -j4

先是报cannot find lnvcuvid的错误，按照一下链接解决了这个问题

<http://blog.csdn.net/lanxueCC/article/details/53422255>

然后

Make clean

Sudo make all –j8

编译成功

Cd 1\_Utilities/deviceQuery

./deviceQuery

如果显示的是一些关于GPU的信息，则说明安装成功了。(最后显示是ＰＡＳＳ不是Faile!)

**6.配置Python环境（有问题好像）**

参考链接：

http://www.linuxidc.com/Linux/2015-07/120449.htm

（1）将终端cd到要安装caffe的位置。

（2）从github上获取caffe：

git clone https://github.com/BVLC/caffe.git

注意：若没有安装Git，需要先安装Git：

sudo apt-get install git

进入python文件夹：

cd python

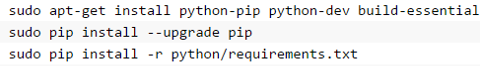
在终端执行下面的命令：

sudo apt-get install python-pip

sudo apt-get install python-numpy python-scipy python-matplotlib ipython ipython-notebook python-pandas python-sympy python-nose

最后安装requirement里面的包，需要root权限

sudo su  
for req in $(cat requirements.txt); do pip install $req; done





**6.编译caffe**

进入caffe根目录， 首先复制一份Makefile.config

cp Makefile.config.example Makefile.config

然后修改里面的内容，

将

#USE\_CUDNN:=1

改为

USE\_CUDNN:=1

MATLAB\_DIR 如果需要使用MATLAB wrapper的同学需要指定matlab的安装路径, 如我的路径为 /usr/local/MATLAB/R2013b (注意该目录下需要包含bin文件夹，bin文件夹里应该包含mex二进制程序)

Make all –j8

Make test

Make runtest

**MNIST数据集测试**

配置caffe完成后，我们可以利用MNIST数据集对caffe进行测试，过程如下：

1.将终端定位到Caffe根目录

cd ~/caffe

2.下载MNIST数据库并解压缩

./data/mnist/get\_mnist.sh

caffe/data/mnist/目录下会多出训练集图片、训练集标签、测试集图片和测试集标签等4个文件

3.将其转换成Lmdb数据库格式

./examples/mnist/create\_mnist.sh

必须要在caffe根目录下运行create\_mnist.sh

此时在/caffe/examples/mnist/目录下生成mnist\_test\_lmdb和mnist\_train\_lmdb两个LMDB格式的训练集和测试集

4.训练网络

LeNet-5模型描述在/caffe/examples/mnist/lenet\_train\_test.prototxt

Solver配置文件在/caffe/examples/mnist/lenet\_solver.prototxt

执行文件在/caffe/examples/mnist/train的lenet.sh

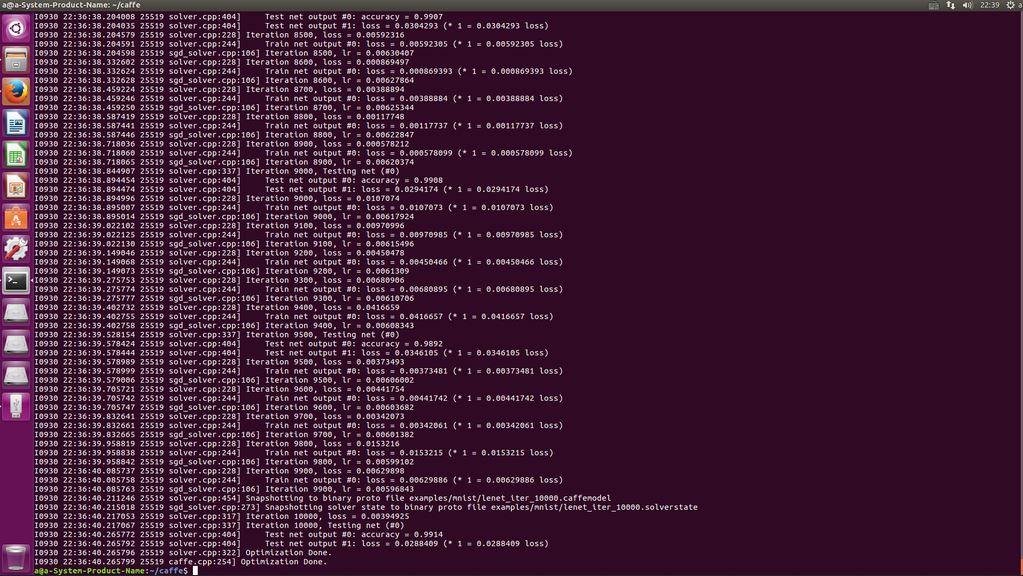
当前位置在caffe根目录下

终端输入

./examples/mnist/train\_lenet.sh

即可开始训练，准确率一般都有0.99以上

训练的时候可以看到损失与精度数值，如下图：



**MATLAB接口配置：**

1.安装matlab2014b

过程见《ubuntu14.04安装matlab\_R2014b.docx》

2.MATLAB接口配置

在caffe目录下的Makefile.config文件中修改matlab的路径：

将

#MATLAB\_DIR:=/usr/local

修改为

MATLAB\_DIR:= /usr/local/MATLAB/R2014b

在caffe目录下输入：

Make matcaffe

（要先编译caffe，即

Make all –j8

Make test –j8

Make runtest –j8）

参考链接：

<http://www.cnblogs.com/platero/p/3993877.html>

配置完成后的结果：

Ubuntu14.04

Gcc4.7

Nvidia 驱动 367.57

Cuda7.5

Cudnn5.0（对应cuda7.5版本）

Jupyter

Matlab2014b

（python好像也可以用，但有问题好像，opencv没有再装3.1的了，但是好像也有opencv）